Developing a web application to improve communication at a software company.

E. Dreyer

Research methodology for the Dissertation submitted in partial fulfillment of the requirements for the degree *Bsc in Information Technology Hons* at the Vaal Campus of the North-West University

Supervisor: Dr. Suné Van Der Linde

Co-supervisor: Mr. Luke Coetzee

Date of submission: 2021/05/12

Version: 1.0

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

EU   European Union (Abbreviation)

LIST OF FIGURES

LIST OF TABLES

No table of figures entries found.

Chapter 3: Literature review

1. Introduction

The goal of this study is to develop a web application that can be used to enhance communication between developers and management at a South African software development company. To reach the goal of this study, an evaluation to improve the communication using different communication methods and user interface design in a software development environment will be done.

According to Kim (2018) doing a literature review sharpens a researchers research focus, as it is the study of past work in order to understand how it relates to the research being done. By doing research on the different communication methods, it opens new possibilities and ideas on how the artefact will improve the communication in the company, as well as understanding the advantages and disadvantage of each communication method. Human-computer interaction helps with understanding of what the difference between a good and a bad system is, thus giving background on designing the artefact, and using past research on user experience.

This chapter addresses the key concepts of the study, namely the importance of good communication as well as the different communication methods used in the industry, followed by human-computer interaction and the importance of a good graphical user interface.

In the next section of the paper the problem description and background will be discussed.

2. Problem description and background

In the corporate world, businesses rely on effective communication to succeed. Developers lack the number of screens that they need to keep all their important tabs open. This makes it harder for important messages to reach developers and influences productivity and creativity (Schrader, 2018).

As a solution, an artifact must be developed to assist with the effectiveness of communication in the industry.

3. Aims and objectives of project

This study proposes the development of a communication web application that can easily be viewed in an office by all employees to allow easy access to important communication regarding specific software development projects. Where the primary objective is to develop a web application for a South African software development company that allows for easy access to important communication relating to specific project.

The theoretical objectives of this study are to identify the communication applications commonly used in industry, as well as gain knowledge of design science research to guide the development of an artifact. Therefore, in the next section of the paper a literature review will be done regarding different communication methods, followed by human-computer interaction.

4. Literature review

4.1. Introduction

According to Rowley and Slack (2004) a literature review is conducting a summary of a subject in a field that promotes the identification of distinct research questions. A literature review needs to gather information using different types of sources, such as books, professional journal articles and websites to find relevant information.

In this study the literature will be used to create a bigger view on the research as the background is explained and what paths was taken to reach the objective of this study. The literature review will ensure further insight in the research field (Knopf, 2006).

The goal of this chapter is to gain insight into the different ways in which communication can take place in the industry, as well as human-computer interaction to ensure that the web application is designed in the most user-friendly way.

In the next section of the study, the importance of good communication is discussed.

4.2. Importance of good communication

It is vital of the importance of good communication within a software development industry is often overlooked. When passing information about code, requirements and bugs it is crucial that information gets passed on correctly to the developers and to the company (Hellgren, 2018). To ensure that the stakeholders get their desired outcome, there needs to be a strong common understanding between the one that is making a request and the person that is completing the request. This common understanding can either lead to improving the project or someone spending time on a feature that is ultimately not what the users wanted, increasing the cost of the project (Dovleac, 2015).

For every scenario in a project, there is an optimal way of communicating. In the next section, the different methods of communicating are discussed.

4.3. Methods of communicating

There are various platforms used for communication within a work setting. According to (Hellgren, 2018), companies use email, instant messaging, wikis, chat systems, issue queues, social media, drawing or multimedia. Each communication method will be discussed in detail in the next section of the paper.

Email – A study exploring the amount of time a person spends reading and responding to emails, reports that an average person uses 28% of their work time reading, replying and writing emails (Hackeling, 2021). It is alarming to consider employees spending more than a quarter of their work time on emails, and in some cases, it involves personal emails as well. To avoid reading unnecessary emails, most companies advise their employees not to use their personal emails for anything work related (Niinimaki, 2011).

Instant messaging – Examples of instant messaging include WhatsApp, Skype, and Slack. Most companies only use one form of an instant messaging system (Leano, 2020), where some companies use more than one, this makes it exponentially harder for important information to reach the desired person (Dittrich & Giuffrida, 2011). The

main disadvantage of instant messaging systems is that people miss messages because they are not always on their phones or at their computers (Hellgren, 2018).

Wikis – Wikis are widely used by many organizations, they are a powerful way of communicating information to a group of people, but lack the functionality to notify that group of people, with important information (LeBar, 2017). Most wikis also do not have good version control for people to see who changes information and at what time (Hellgren, 2018). Information also need to be kept up to date, or else they do not benefit the project (Minocha et al., 2008).

Chat Systems – Chat applications like Rocket.chat, IRC and LiveAgent. Chat systems are great for sharing a lot of information to a group of people (Phpzag, 2021). They also do not overload the users when sharing information, like an Instant messaging system, because users can read the messages on their own time and concentrate on what is being said (Hellgren, 2018). A drawback of this form of communication is that you do not always want people to read the information on their own time, you want them to work on the feature or bug when you ask them to do so. They also lack notifications when new content is uploaded.

Issue queues – Examples of issue queues include Bugzilla, Jira, or Trello to exchange information about development. The problem with this is the lack of feedback, when someone completed a feature or bug, as they are just moved around to the specific topic with no explanation on why it is there (Zepel, 2021). This communication method brings a lot of value to the development process, this is useful as a side feature for this study and will definitely help solve the main objective (Bérczes et al., 2012).

Social media – Many social media platforms, for example Workplace by Facebook, have tools that allow companies to have their own channels for employees (Hellgren, 2018). Each employee has their own voice, and seeing employees work on problems in their own way. The downside of using this method is that the company pays for the brand. This making it more expensive than for example using an issue queue, and with social media platforms selling their user data for advertising, the security is not the best if you are discussing private business related issues (Blaszkiewicz, 2017).

Drawing – Drawing rich pictures is a great way to get employees to see things in new perspectives (Elin, 2012). The problem with this is combining verbal communication because you are not always in the same room or using another communication method (Cockburn, 2006). Same goes for making a PowerPoint or digital images. There is also poor version control as you cannot go back to older versions of the drawing (Hellgren, 2018).

Multimedia – methods include the use of audio recording or videos and is perfect for sharing media files to multiple people. This is beneficial for making training videos, so that you only have to explain something once, the problem is keeping the videos up to date, and people also find it harder to ask questions when they are watching a video compared to someone explain to them in person (Sarowardy, 2019).

4.3.1. Conclusion

There are more communication methods, but these are the most popular amongst software developing companies (Hellgren, 2018). All of them have their own advantages and disadvantages and can cause developers to spend time on things that are necessary for the development of the project, or waste time on something that does not add to the completion of the project (Storey et al., 2016).

4.4. Interruptions

According to Thorne (2020) it takes a person an average time of 25 minutes (to be exact, 23 minutes and 15 seconds) to return to a task after being interrupted. This can be frustrating to developers, and lost time can cost the company money. To be constantly interrupted, can cause a decrease in job satisfaction and lead to performance issues.

For this study, a method that would be most applicable is one that does not immediately interrupt a developer, while still being notified that there is important information that needs to be attended to. This will allow the developer or project manager to continue with what they are busy with and allow them to make it part of their routine to check their notifications.

4.5. Human-computer interaction (HCI)

According to Hinze-Hoare (2007) HCI is the way in which an artifact is designed to provide the best user experience. For one to achieve the best user experience, you need to understand the interaction between the system and the user (Terblanche, 2014). Human-computer interaction helps with understanding what the difference between a good and a bad system is, but it does not guarantee the development of a successful product (Blackwell, 2010).

Hinze-Hoare (2007) has produced ten rules that should be followed by all interface designers to guide them to create the best user experience. In the next section of the paper, the ten rules will be listed and explained in detail, with example of how it should be done in the study.

4.5.1. Simple and natural dialogue

According to Chen (2018), there should not be any sight of irrelevant information, Chen (2018) described it as having a "signal-to-noise" ratio. "Signal" being good because the user can easily find what they want to achieve in the system, whereas "noise" being bad because the system is giving the user irrelevant information that does not contribute to achieving their goal.

According to Terblanche (2014) every extra information on the screen competes for visibility. In a graphical-user-interface the "information" that is displayed can be anything from visual elements, text content, or animation (Chen, 2018). Not every user wants to achieve the same goal, making the signal-to-noise ratio change depending on the user.
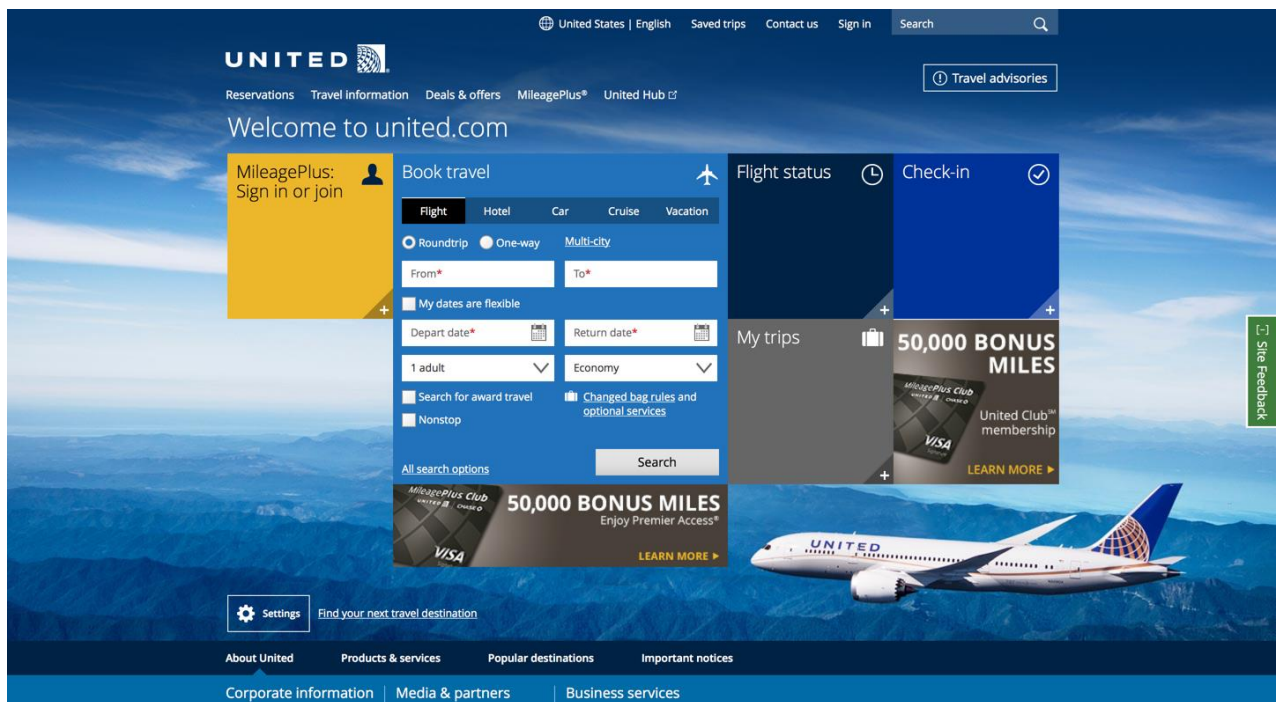
Figure 1: Simple and natural dialogue (Chen, 2018)

As shown in Figure 1, for a user that wants to book a new flight, the "Book travel" is a strong enough signal for them to easily find what they are looking for. Whereas a user that wants to see their flight status, the "Book travel" block is in the way because it is taking up most of the space (Chen, 2018).

4.5.2. Speak the users' language

The information that is available to the user should be expressed so that the user understands, and is familiar to the concepts, rather than what is understood by the system or the operator (Foundation, 2016).

Keep the language simple, and do not explain terms that people are unfamiliar with. For example a user that speak English as a second language will not necessarily understand an idiom and will need an explanation because they are unfamiliar with the concept (Foundation, 2016).
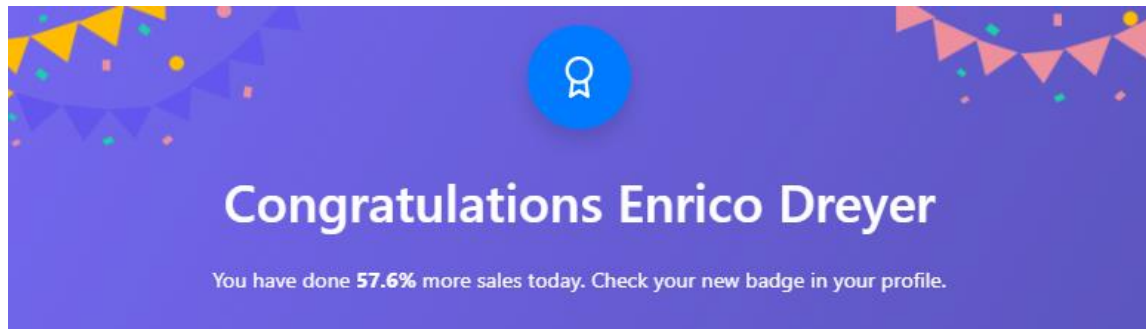
The language being used includes dialog boxes, menus, help files and instructions (Bridge, 2018). This can also include changing the language of the whole system, to give the user a better experience in their own language.

4.5.3. Minimize the users' memory load

According to Nielsen (2001), the importance here is to not have the user remember information from one page to the other to carry out tasks, and the user's information should be easily retrievable. By adding memory load, the users' capability to perform their main task gets reduced.

Ways in which memory load can be reduced are using a menu instead of commands, using default values, using visualization as external information, examples for the user and generic actions and rules (Nielsen, 2001).
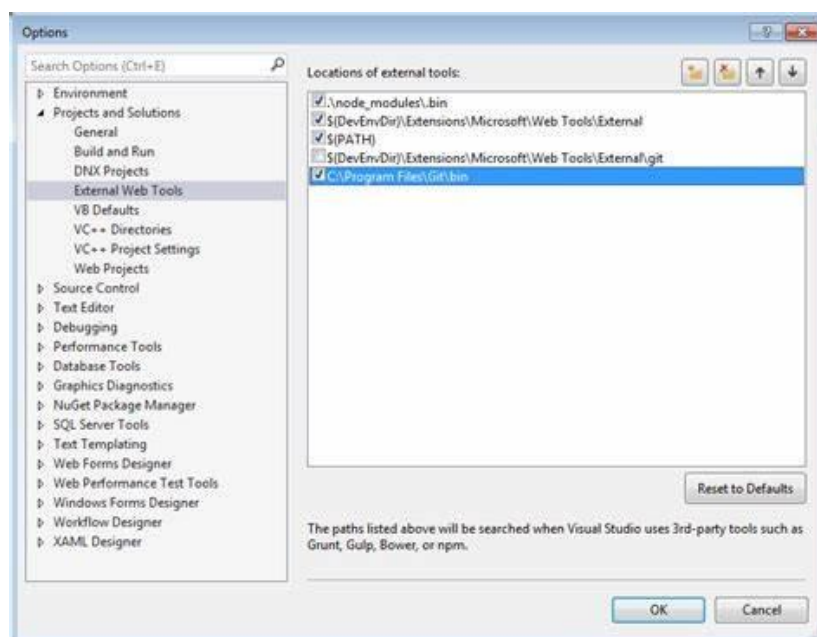
As shown in Figure 3, the user can select options instead of running commands and default values are selected to minimize confusion. Another example is when an administrator clicks on one user profile, he does not want to remember the users name before he changes it. The information should already be on the page ready for the user to edit or confirm detail.

4.5.4 Consistency

This means that all generic buttons should be on the same place no matter where they occur in the system. According to Beckert and Beuster (2006), if there is a lack of consistency it can lead wrong interpretation of data or insufficient information given to the user. The system might provide a lack of information to determine the system state, or the user might interpret the output of the system wrongly.
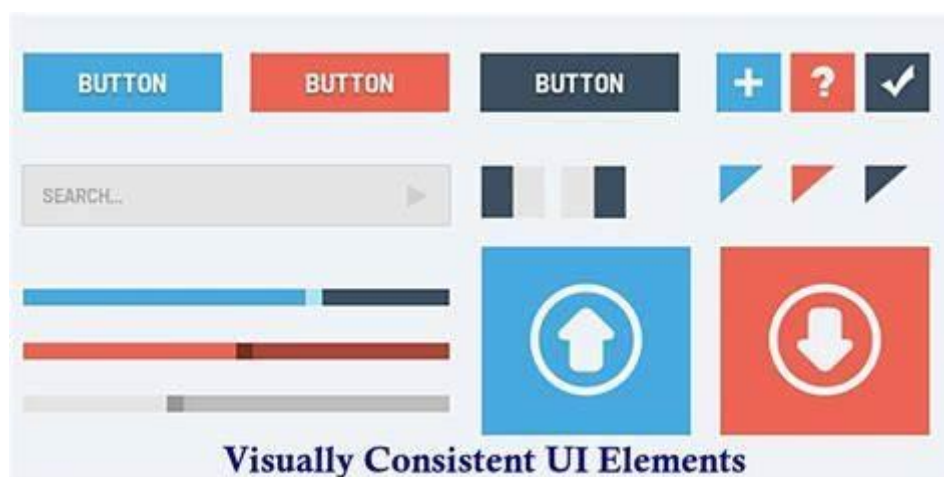

Visually Consistent UI Elements

Figure 4: Consistency (Anić, 2015)

This is also applicable to "save", "delete", and "cancel" buttons that should always be on the bottom of the page as well as having the same colour all over the system, for example delete should always be red and save should always be green, this increases usability throughout the system (Anić, 2015).

4.5.5. Feedback

The user should always be aware of what is going on in the system, in a relevant and timely manner (Natoli, 2020). For example, when the user finished making changes and

clicks on "save", there should be a feedback notification stating that they did everything correctly and their changes have been successfully saved, or that their changes have been unsuccessful because the system is down.

According to Natoli (2020) the feedback given by the system should answer questions from these four categories:

- Future Status: What will happen next?

- Current Status: What is happening?

- Location: Where am I?

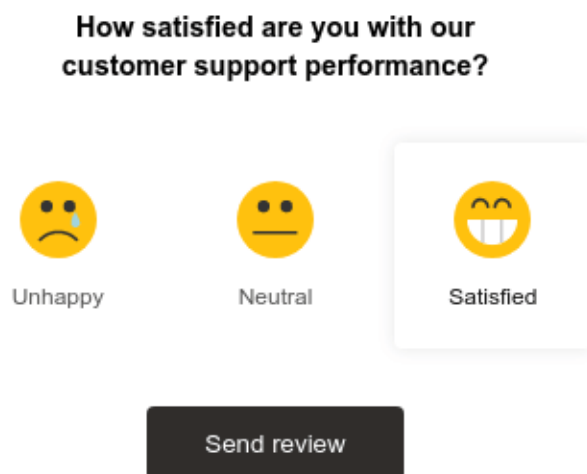- Outcome and Result: What just happened?

Figure 5: Feedback (Natoli, 2020)

Above all else, the feedback given by the system is there to provide answers to the questions that the human brain instinctively asks, such as "s the work saved?", "Am I finished?" and "What should I click?" (Natoli, 2020). This is also applicable to loaders that show that the page is still loading, or that the changes are being applied to the database.

4.5.6. Clearly marked exists

According to Blender (2015) errors can be easily made by a user, thus they need an easy way to return to the previous state without having to proceed with their mistake. The simplest way to display an error to a user, is to have a text box display over the process that is currently happening (Blackwell, 2010). This will inform the user that something went wrong, and they need to return to the previous state of the system.
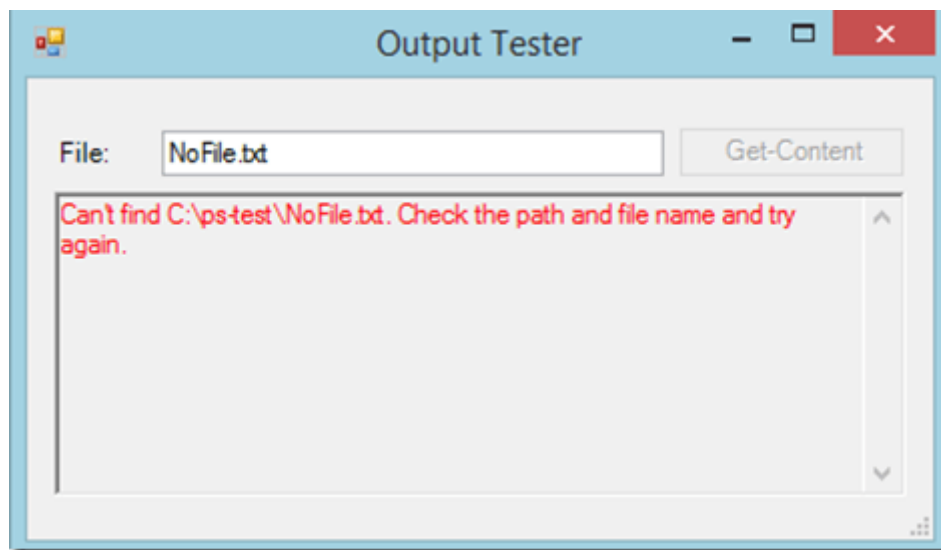
Figure 6: Clearly marked exists (Blender, 2015)

Another example of clearly marked exits other than the one displayed in Figure 6, is to give the user the option to return to the previous state. On every page, where you can edit or add data, there should be a "cancel" or "reset" button, that takes you back to the previous state. Consistency also plays a role, as exists must be on every page at the same place, so that the user always knows how to exit the page (Blackwell, 2010).

4.5.7. Shortcuts

According to Keerti (2020), shortcuts is used to speed up the interaction with the system by a user. This can be done by implementing a menu, using icons to represent a word, extra buttons or windows.

By adding a limited number of options to the menu, when the user selects the menu, it opens a sub-menu that gives further information. Menu-driven user interface helps with

the process of having to continue with the main page to reach your goal, but rather skipping to the desired goal (Keerti, 2020).
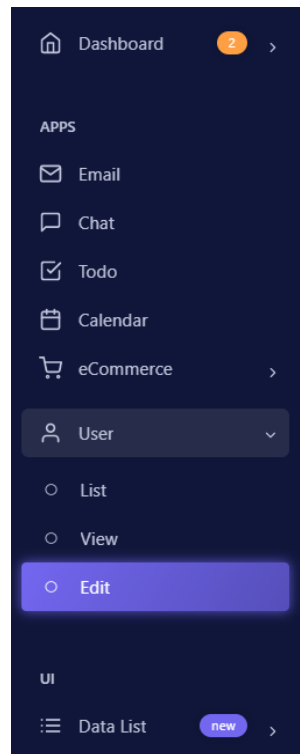
Figure 7: Shortcuts (own example)

As shown in Figure 7, there are also different menu levels, such as "List" users, "view" users, and "Edit" users. This makes it easier for experienced user to get to their desired outcome faster and gives the user insight on what are other outcomes of that main menu (Ngai, 2017).

4.5.8. Good error messages

Error messages should be displayed in plain language for the user to understand, and user-friendly error messages should answer the "what?", "why?" and "how?" (Gregory, 2019):

- What is the problem?

- Why did the problem occur?

- How do I solve the problem?

According to Gregory (2019), error messages need to be:

1. Specific to the user's task.

2. Let the users think that the system thinks like a human.

3. The possibility for humour in the situation.

4. Do not make users work for their desired outcome.

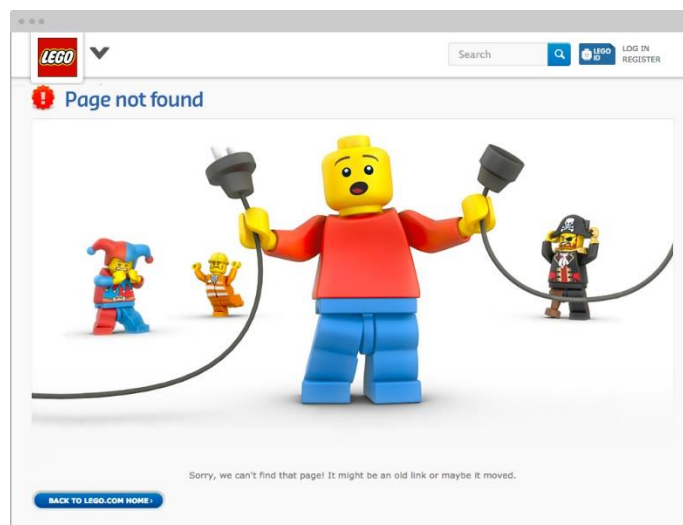5. Have a useful "page not found" page.

This also falls under consistency and feedback, as this should always show up at the same place, when the user did something wrong (Beckert & Beuster, 2006).

4.5.9. Prevent Errors

According to Continelli (2017), companies that rely on their systems to function all of the time, an error in the system can lead to financial loss, and the main cause of system failure is human error.

Human error can be prevented by carefully designing what the user has access to. This can be done by setting the user permissions and having them see and edit only allowing them the data that is relevant (Laubheimer, 2015). Users do not realize when

they are about to do something that can trigger the system to fail, thus giving them a preview of the results can lower the chances of error (Laubheimer, 2015).
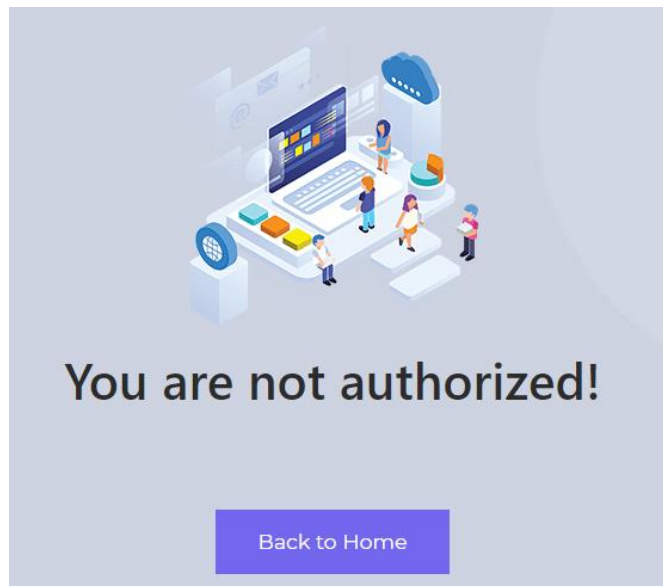

You are not authorized!

Back to Home

Figure 9: Prevent Errors (own example)

As shown in Figure 9, bugs do occur, and users get access to things that they should have access to (Laubheimer, 2015). This can be prevented by setting up the backend to look at the users' permissions and when the user does not have access to the page it redirects to the page shown in Figure 9.

4.5.10. Help and documentation

According to Trica (2019), if your system speaks for itself, it can be used without documentation, but if the users need help with anything in the system, it can be easy to allocate. There should also be documentation for the users and developers, this can impact the further development of the project as well as give users background of the system.

Included in the documentation for developers, there should be:

- Server Environments

- Business rules

- Database files

- Trouble shooting

- Application installation

- Code

Whereas the user documentation should be focused on the users' tasks and the steps to achieving specific solutions (Trica, 2019).
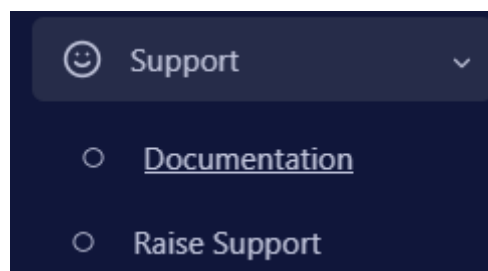
4.6 HCI Conclusion

According to Fawcett (2021), it is important not to reinvent the wheel, but still use your own innovation. Developers have the tendency to transfer old interfaces to well-designed systems, and to prevent that a design that has set standards, prove to be more intuitive and have all human-computer interaction set in place (Fawcett, 2021). For this study, all human-computer interaction rules will be followed to improve on user experience.

The next section of the paper will be the summary of this chapter.

5. Summary

To reach the goal of this study, an evaluation to improve the communication using different communication methods and user interface design in a software development environment will be done.

The theoretical objectives of this study are to identify the communication applications commonly used in industry, as well as gain knowledge of design science research to guide the development of an artifact. By doing research on the different communication

methods, it opens new possibilities and ideas on how the artefact will improve the communication in the company, as well as understanding the advantages and disadvantage of each communication method. Human-computer interaction helps with understanding of what the difference between a good and a bad system is, thus giving background on designing the artefact, and using past research of user experience.

According to (Hellgren, 2018), companies use email, instant messaging, wikis, chat systems, issue queues, social media, drawing or multimedia. In this chapter each method was discussed in depth.

For this study, a method that would be most applicable is one that does not immediately interrupt a developer, while still being notified that there is important information that needs to be attended to. This will allow the developer or project manager to continue with what they are busy with and allow them to make it part of their routine to check their notifications.

Human-computer interaction helps with understanding what the difference between a good and a bad system is, but it does not guarantee the development of a successful product (Blackwell, 2010). All human-computer interaction rules will be followed to improve on user experience.

The next chapter in this study will be the empirical study and will cover the data gathering techniques used to analyse and improve the existing concept.

6. Reference List

Anić, I. (2015). The importance of Visual Consistency in UI Design. https://www.uxpassion.com/blog/the-importance-of-visual-consistency-in-ui-design/

Beckert, B., & Beuster, G. (2006). Guaranteeing consistency in text-based human-computer-interaction. *FMIS 2006*, 57.

Bérczes, T., Sztrik, J., & Orosz, P. (2012). Tool supported modeling of sensor communication networks by using finite-source priority retrial queues. *Carpathian Journal of Electronic and computer engineering*, *5*, 13.

Blackwell, A. (2010). Human Computer Interaction. https://www.cl.cam.ac.uk/teaching/1011/HCI/HCI2010.pdf

Blaszkiewicz, S. (2017). Can Workplace by Facebook, work? https://www.getapp.com/resources/can-facebook-at-work-work/

Blender, J. (2015). Manage Errors in a GUI Application. https://www.sapien.com/blog/2015/01/15/manage-errors-in-a-gui-application/

Bridge, K. (2018). User Interface Language Management. https://docs.microsoft.com/en-us/windows/win32/intl/user-interface-language-management#:~:text=The%20user%20UI%20language%20determines%20the%20user%20interface,user%20interface%20language%20is%20installed%20on%20the%20computer.

Chen, X. (2018). Signal–to–Noise Ratio. https://www.nngroup.com/articles/signal-noise-ratio/#:~:text=Definition%3A%20In%20human%E2%80%93computer%20interaction%2C%20the%20signal%E2%80%93to%E2%80%93noise%20ratio%20represents,anything%20%E2%80%94%20text%20content%2C%20visual%20elements%2C%20or%20animation.

Cockburn, A. (2006). *Agile software development: the cooperative game*. Pearson Education.

Continelli, A. (2017). How to Identify and Prevent Software Failure Risks. https://www.business.com/articles/aaron-continelli-identify-and-prevent-software-failure/

Dittrich, Y., & Giuffrida, R. (2011). Exploring the role of instant messaging in a global software development project. 2011 IEEE Sixth International Conference on Global Software Engineering,

Dovleac, L. (2015). The role of new communication technologies in companies' sustainability. *Bulletin of the Transilvania University of Brasov. Economic Sciences. Series V*, *8*(1), 33.

Elin, S. (2012). Drawing and Painting as a Tool for

Communication. https://www.duo.uio.no/bitstream/handle/10852/37146/NewMasterThesistemplate%5B2%5D-8.pdf?sequence=1

Fawcett, A. (2021). Introduction to Human-Computer Interaction & Design Principles. https://www.educative.io/blog/intro-human-computer-interaction

Foundation, I. D. (2016). The Use of Language and the User Experience. https://www.interaction-design.org/literature/article/the-use-of-language-and-the-user-experience

Gregory, S. (2019). Best Error Messages: 5 Tips For A User-Friendly Experience. https://freshsparks.com/user-experience-tips-best-error-messages/#:~:text=A%20more%20positive%20error%20message%20experience%20is%20to,users%20engaged%20and%20willing%20to%20make%20the%20corrections.

Hackeling, E. (2021). How much time are you spending on email? . https://frontapp.com/blog/how-much-time-are-you-spending-on-email

Hellgren, H. (2018). Communicating in software development. https://hackernoon.com/communicating-in-software-development-f3434c52eb23

Hinze-Hoare, V. (2007). The review and analysis of human computer interaction (HCI) principles. *arXiv preprint arXiv:0707.3638*.

Keerti. (2020). *A Complete Guide for UI Design Process (User Interface)*. https://www.cronj.com/blog/user-interface-ui-design-process-in-graphic-design

Kim, Y. S. (2018). The Importance of Literature Review in Research Writing. https://owlcation.com/humanities/literature_review

Knopf, J. W. (2006). Doing a literature review. *PS: Political Science and Politics*, *39*(1), 127-132.

Laubheimer, P. (2015). Preventing User Errors: Avoiding Conscious Mistakes. https://www.nngroup.com/articles/user-mistakes/#:~:text=Prevent%20mistakes%20by%20helping%20the%20user%20to%20build,before%20deleting%29%2C%20and%20warn%20before%20mistakes%20are%20made.

Leano, M. (2020). WHY INSTANT MESSAGING IS THE TOP COMMUNITION TOOL FOR BUSINESSES. https://newmediaservices.com.au/instant-messaging/#:~:text=Instant%20messaging%2C%20originally%20created%20for%20entertainment%20and%20social,messaging%20marketing%2C%20lead%20generation%2C%20and%20customer%20support%20purposes.

LeBar, Z. (2017). What Are Wikis, and Why Should You Use Them? https://business.tutsplus.com/tutorials/what-are-wikis-and-why-should-you-use-them--cms-19540

Minocha, S., Petre, M., & Roberts, D. (2008). Using wikis to simulate distributed requirements development in a software engineering course. *International Journal of Engineering Education*, *24*(4), 689-704.

Natoli, J. (2020). Feedback: 5 Principles of Interaction Design To Supercharge Your UI (5 of 5). https://givegoodux.com/feedback-5-principles-interaction-design-supercharge-ui-5-5/

Ngai, J. (2017). Important Advantages of Data Driven Design. https://webdesign.tutsplus.com/articles/important-advantages-of-data-driven-design--cms-29115

Nielsen, J. (2001). 113 Design Guidelines for Homepage Usability. https://www.nngroup.com/articles/113-design-guidelines-homepage-usability/

Niinimaki, T. (2011). Face-to-face, email and instant messaging in distributed agile software development project. 2011 IEEE Sixth International Conference on Global Software Engineering Workshop,

Phpzag, T. (2021). Build Live Chat System with Ajax, PHP & MySQL. https://www.phpzag.com/build-live-chat-system-with-ajax-php-mysql/#:~:text=Chat%20System%20or%20Chat%20application%20is%20mainl

y%20used,to%20assist%20them%20regarding%20services%20and%20resolve%20issues.

Rowley, J., & Slack, F. (2004). Conducting a literature review. *Management research news*.

Sarowardy, M. H. (2019). The Issues and Challenges of Using Multimedia at a District Level. https://www.scirp.org/html/11-6304491_93691.htm

Schrader, J. (2018, 30 July). *How Your Cell Phone Habits Impact Your Productivity*. https://www.psychologytoday.com/us/blog/why-bad-looks-good/201807/how-your-cell-phone-habits-impact-your-productivity

Storey, M.-A., Zagalsky, A., Figueira Filho, F., Singer, L., & German, D. M. (2016). How social and communication channels shape and challenge a participatory culture in software development. *IEEE Transactions on Software Engineering*, *43*(2), 185-204.

Terblanche, J. T. (2014). Using HCI principles to design interactive learning material. *Mediterranean Journal of Social Sciences*, *5*(21), 377.

Thorne, B. (2020). How Distractions At Work Take Up More Time Than You Think. http://blog.idonethis.com/distractions-at-work/#:~:text=It%20takes%20an%20average%20of%20about%2025%20minutes,University%20of%20California%2C%20Irvine.%20Multiple%20studies%20confirm%20this.

Trica, A. (2019). The Importance of Documentation in Software Development. https://filtered.com/blog/post/project-management/the-importance-of-documentation-in-software-development

UI, K. (2021). Kendo UI Core on Bower. https://docs.telerik.com/aspnet-core/installation/bower-install

Zepel. (2021). *8 Problems You'll Face By Using Trello for Agile Web Development*. https://zepel.io/blog/problems-using-trello-for-agile-web-development/